

プログラムの概略

平成 28 年 9 月 4 日

1 歴史

1.1 開発に関係した人々

xTAPP は慶應義塾大学の山内淳氏が開発してきた平面波基底第一原理計算プログラム群 TAPP の version 1.0.8 から 2000 年ごろ吉本芳英が分岐させたものである。

TAPP は、平面波基底擬ポテンシャル全エネルギー計算プログラム PPSF を基礎骨格として、東京大学工学部 押山淳氏のグループの東京大学 物性研究所 杉野修氏が開発した共役勾配法による対角化コードを取り込んで、山内氏が ultrasoft 擬ポテンシャルに対応させたプログラムが中心となっている。その経緯は、1990 年頃に塚田 捷 東京大学名誉教授の下、物質・材料研究機構 大野隆央氏、杉野氏、東京大学工学部 渡邊聡氏、山内氏が協力して開発を開始し、基本設計を杉野氏、バンド計算部分、擬ポテンシャル作成部分の coding は、それぞれ主に山内氏、渡邊氏が担当した。

PPSF は、鹿児島大学 永吉秀夫氏が開発した経験的局所擬ポテンシャル計算プログラムを参考に、名古屋大学工学部 白石賢二氏が一から coding した行列対角化による semilocal の非経験的擬ポテンシャル用平面波基底全エネルギー計算プログラムで、高度に対称性を利用した効率的アルゴリズムを特長としている。

PPSF/TAPP は、主として上記のメンバーによるコードからなるが、当時から現在まで、東京大学 理学部 物理学科 植村、上村、塚田、青木、常行研究室の関係者が直接間接に大きく寄与している。特に、兵庫県立大 島信幸氏、千葉大学 中山隆史氏 (PPSF/TAPP)、当時大学院生であった影島博之氏、木村栄伸氏、東北大学 赤木和人氏、東京理科大 (当時) 諏訪雄二氏 (TAPP) にはお世話になっている。

xTAPP 一般公開版計画には、東京大学理学系研究科 常行真司氏のご協力をいただいている。2012 年夏から作業が始まった xTAPP 一般公開版の新入力形式のプログラムは常行研究室の吉澤香奈子氏によるものが基礎になっている。また、山内氏、日立製作所 諏訪雄二氏にも作業に加わっていただいた。

1.2 開発に関係した資金など

xTAPP には、文部科学省科学研究費補助金「新学術領域」平成 22 年度～平成 26 年度コンピュータによる物質デザイン：複合相関と非平衡ダイナミクス、による成果が含まれている。

2 関係するプログラム

擬ポテンシャル作成は日立製作所 諏訪雄二氏が保守しているプログラム群 psv を xTAPP 向けに変更したものが必要であるがこのパッケージには含まれていない。擬ポテンシャルで原子を解

くプログラム solps、solspin はこのプログラム群に含まれている。

3 主な機能

- 平面波基底による第一原理計算
- Ultrasoft pseudo-potential
- semi local および hybrid 交換相関汎関数
- LDA(スピン偏極なし) と LSDA(スピン偏極あり)
- non linear core correction (partial core charge)
- 固有エネルギー、電荷分布、波動関数、原子に働く力、セルのストレス
- 構造最適化。セル固定とセル可変。
- エネルギー障壁。hyper plane constraint 法と force inversion 法。
- 第一原理分子動力学 (BOMD)、NVE、NPT。
- バンド図、軌道電荷分布の積分値
- total DOS、projected DOS、電荷分布の積分値のエネルギー分解
- Γ 点計算および反転対称性がある場合の計算に対する計算の最適化
- OpenMP および MPI 並列化
- hybrid 汎関数については GPGPU 対応
- OpenDX[1] での電荷密度、波動関数などの可視化
- 大谷、杉野による ESM[2] 機能
- 九工大の中村による最局在ワニエ関数の計算機能。応用例は [3,4]。解説は [5]。
- S. Grimme による DFT-D 計算 [6,7,8]
- phonopy[9] を用いたスーパーセル法によるフォノンの計算。
- スピン軌道相互作用および non-collinear 磁性。

[1] <http://www.opendx.org>

[2] M. Otani and O. Sugino, Phys. Rev. B 73, 115407 (2006)

[3] K. Nakamura et al., Phys. Rev. B 74, 235113 (2006)

[4] K. Nakamura et al., J. Phys. Soc. Jpn 77, 093711 (2008)

[5] 中村和磨, 表面科学 29, 432-436 (2008)

[6] S. Grimme et al. J. Chem. Phys, **132** 154104 (2010)

[7] S. Grimme et al. J. Comput. Chem. **27** 1787-1799 (2006).

[8] S. Grimme et al. J. Comput. Chem, **32** 1456-1465 (2011).

[9] <http://phonopy.sourceforge.net/index.html>

4 プログラムの概略

このパッケージには以下のプログラムが含まれている。

inipot 入力データから平面波基底を生成し、擬ポテンシャルを平面波基底で表現するデータを生成する初期化プログラムである。pefcos を除くその他のプログラムはこのプログラムで生成した初期化データを必要としており、最初に動かす必要がある。ただし、データその物は計算条件を固定すれば使いまわしできる。

cgmrrpt 構造最適化を行うプログラムである。

vbpef cgmrrpt で計算したローカルポテンシャルと波動関数 (hybrid 汎関数の場合) で決まるハミルトニアンを固定して、逆空間上に設定する k 点の軌跡の上で、固有エネルギー、波動関数、軌道電荷分布の空間積分値、軌道電荷分布そのものを求めるプログラムである。いわゆるバンド図のデータを生成するのはこのプログラムである。

vbstm cgmrrpt で計算したローカルポテンシャルと波動関数 (hybrid 汎関数の場合) から STM 像のシミュレーションを局所状態密度の積分値を用いて行うプログラムである。

wfn2chg cgmrrpt で計算した波動関数から、電荷分布の空間積分を固有エネルギーで分解したものや projected DOS を計算するプログラムである。

mdrrpt 第一原理分子動力学 (BOMD) を行うプログラムである。

wannier cgmrrpt で計算した波動関数から、最局在ワニエ関数を計算するプログラムである。

pefcos cgmrrpt で計算されているバンドの cos 展開データからバンド図を生成するプログラムである。

pe2dos cgmrrpt で計算されているバンドの cos 展開データから状態密度を計算するプログラムである。

tetrapdos wfn2chg が出力するファイルから projected DOS を計算する。

xticonv 入力ファイルの構造データを可視化ツール用に変換するツール

strconv 構造最適化の結果ファイルを可視化ツール用に変換するツール

diffstr 構造最適化の結果ファイルを二つ受け取って、その構造データの差分ベクトルをセル座標で出力するツール

iplstr 構造最適化の結果ファイルを二つ受け取って、その構造データを線形補間した構造データをセル座標で出力するツール。

cmpstr 構造最適化の結果ファイルが誤差の範囲で同一かチェックするツール

入力データの詳細については inputformat.tex を参照すること。なお、入出力においてはとくに断らない限り原子単位系が使われる。

また、プログラムの定式化については formalism.tex を参照すること。

5 プログラムのコンパイルについて

5.1 はじめに

コンパイルのための設定が Makefile に用意されているのは Intel 環境と京コンピュータである。この他の環境で性能がでるようにコンパイルしたい場合には開発者に問い合わせるべきである。他の環境用の設定を編集して作成することは容易ではない。

コンパイルには MPI が必要である。Intel 環境では OpenMPI で開発が行われている。MPI-2 の機能は利用していない。ライブラリとして BLAS と LAPACK を利用する。Intel 環境では FFT を MKL で実行させるべきである。京コンピュータの場合 FFTW を用いるのがよい。

プログラムのコンパイルはソースのある `src` ディレクトリで行う。

5.2 コンパイルのための各種設定

Makefile-dist

から

Makefile

を生成する。また通常必要は無いが、利用目的によっては `config.h`、`config90.h` で定義されている最大限保持できる対称操作の数 `nsymq` を大きくする必要がある。

Makefile にある、`config` の例を編集して必要な設定を行い、`make` する。FFT ルーチン、LAPACK、密行列対角化ルーチン、交換相互作用の計算コードなどの選択を行うこと。

5.2.1 FFT

FFT3D_OBJ を選択する。FFT が使用するデータフォーマット (`interleave` または `planar`) に合わせて FFTGRID_H_SRC を選択しなければならない。 `fftgrid-scl.h` は `interleave` 用、 `fftgrid-vec.h` は `planar` 用である。MKL と FFTW は `interleave` と組み合わせること。

- MKL 用の `fft3d_mkl.o`
- FFTW 用の `fft3d_fftw.o`
- xTAPP 同梱の `pfft3duz.o` など

5.2.2 密行列対角化ルーチン

EIGSYSTEM_OBJ を選択する。

- xTAPP 同梱の `eigsystm.o`
- LAPACK の `eigsystm-lapack.o`
- ScaLAPACK の `eigsystm-scalapack.o`

を選択する。最適化されたライブラリがあるならそれを使用する。intel MKL は高速である。

5.2.3 交換相互作用の計算コード

交換相互作用の計算コードには GPGPU に対応したものが用意されている。Intel 向けの config の場合、

```
# for NVIDIA CUDA
...
# for AMD GPU
...
# for CPU
...
```

となっている所からそれぞれ、NVIDIA CUDA を使う場合の設定、AMD GPU を使う場合の設定 CPU を使う場合の設定が選択できる。またこれらに合わせて SCGDG0BJ の選択も行うこと。

京コンピュータ向けの config の場合は CPU のみ選択できる。SCGDG0BJ の選択では CPU を選択すること。

5.3 MKL の LAPACK

MKL の LAPACK のライブラリのファイル名は composerxe-2011 で `-lmkl_lapack` から `-lmkl_lapack95_lp64` に変更になっているので注意すること。

5.4 ビルド

オプションなしで `make` すると、デフォルトですべてのプログラムが作成される。プログラムは `src` ディレクトリ中にできるので、必要であればこれを適切な場所にコピーして使う。

6 各プログラムの説明

6.1 inipot

inipot は入力データから平面波基底を生成し、擬ポテンシャルを平面波基底で表現するデータを生成する初期化プログラムである。pefcos を除くその他のプログラムはこのプログラムで生成した初期化データを必要としており、最初に動かす必要がある。ただし、データその物は計算条件を固定すれば使いまわしできる。固定されるべき条件が固定されていない場合、自動的に後続のプログラムは停止する。固定されるべき条件は、

- 平面波基底の数を決めるカットオフと \cos 展開の係数の数を決めるカットオフ。
- 単位胞の形状 (単位胞を構造最適化している場合は基準となる単位胞。)
- 対称性
- サンプル k 点
- 原子種の数と原子の数

である。

このプログラムによって、平面波基底の数が決定される。このプログラムの入力に使った格子パラメータが基準となる。

このプログラム群では擬ポテンシャルの動径データから擬ポテンシャルの逆空間表示を連続関数としてフィッティングしたものを用いるが、これを inipot は計算している。

inipot は論理機番 10 から入力データを読み込む。MPI 並列実行できるが、MPI プロセスのうち、rank 0 のみが実際の処理を行い、他のプロセスは待っているだけである。

inipot は擬ポテンシャルの入力データを必要とするが、一番目の擬ポテンシャルは論理機番 34 から二番目は論理機番 34+ 1 からと言うように順に読み込まれる。この他、論理機番 28 から一番目の擬ポテンシャルに対応する原子電荷分布データ、論理機番 28+ 1 から二番目の擬ポテンシャルに対応する原子電荷分布データと順に原子電荷分布データを順に読み込ませることもできる。ここで読んだ原子電荷分布は初期ローカルポテンシャルを作成する時に使用させることができる。

論理機番 28 などに与えるファイルは、擬ポテンシャルを用いて原子を解くプログラム (solps、solspin) を用いて計算することができる。なお、このファイルには spin 偏極を含めることもでき、初期条件を偏極させたい場合に使用する。

また論理機番 13,14,15,40,22,73,74 に出力を行う。この他、論理機番 21 にも出力を行うが後方互換性のためである。内部のデータは使用されていない。

6.1.1 replica 実行

cgmrpt 同様、replica 実行に対応する。詳細は cgmrpt での説明を参照すること。

6.2 cgmrpt

cgmrpt は電子状態計算を行い、必要なら構造最適化を行うプログラムである。

6.2.1 電子状態計算

電子状態計算は周期境界条件の元で行われる。そのため適切なサンプル k 点を指定する必要がある。部分並進を含めた対称性が考慮されて既約なサンプル k 点を選出され計算は既約なサンプル k 点のみで行われる。

電子状態計算はいわゆる SCF ループの形式で行われる。1 SCF は固定されたハミルトニアンのもとで Davidson 法によって実行される逐次対角化である。

SCF ループで自己無撞着にされるローカルポテンシャルは Anderson extrapolation 法 [1] によってその収束を加速されている。Anderson extrapolation 法の混合パラメータは 0 から 1 までの実数で 0 に近いほど減速が強くなり SCF の不安定性を抑えられるが収束その物は遅くなる。

軌道への電子の詰め方を決める Fermi 面の smearing のアルゴリズムには、cos 展開、Methfessel-Paxton 法 [2,3]、Fermi 関数の三種類を使える。電子系の設定温度はこのアルゴリズムのパラメータである。また、系の中性状態から追加で電子を加えたり、全スピン偏極値を固定したりできる。

non-collinear 計算の場合、交換相関エネルギー E_{xc} と電子密度 $\rho(\mathbf{r})$ 、スピン密度ベクトル $\mathbf{m}(\mathbf{r})$ の関係は、入力に noncol_spin_axis を指定しない場合

$$E_{xc} = \int f_{xc}[\rho(\mathbf{r}), |\mathbf{m}(\mathbf{r})|] d\mathbf{r}$$

であるが、指定する場合には対応する単位ベクトルを \mathbf{u} として

$$E_{xc} = \int f_{xc}[\rho(\mathbf{r}), \mathbf{u} \cdot \mathbf{m}(\mathbf{r})] d\mathbf{r}$$

となる。つまり近似交換相関汎関数に渡す分極度の定義が変化する。後者は collinear になる系に対して用いることが想定されている。交換相関汎関数に PBE を用いる場合、後者の設定にしないと SCF が収束しないことがある。

6.3 原子構造の最適化

原子構造の構造最適化は共役勾配法を用いて行われる。構造最適化は、原子数 N について $3N$ 次元空間内のベクトルで表せる探索方向（共役勾配方向）について次元構造最適化を行う主ステップとその探索方向を更新する副ステップの二重ループで実行されている。

構造最適化においては個別の原子ごとに固定条件を課することができる。指定した系の対称性は自動的に保持される。これらは、計算した力からこれらを破る成分が取り除かれることで実現される。収束判定などの計算はこの処理済みの力について行っている。

また、セル形状の最適化を行えるがこの場合には、電子の運動エネルギーを修正して実質的な基底のカットオフエネルギーを一定に保つ BTP[4] による運動エネルギー補正を使用するべきである。

また構造最適化の変種として、hyper plane constraint 法と force inversion 法 [5] を用いることができる。

hyper plane constraint 法では、原子の動く方向は与えた初期構造をとり、指定した法線ベクトルをもつ超平面の上に制約される。したがって同一の法線ベクトルに対して、切片を動かしながら初期構造を与えてそれぞれ構造最適化し、その最適化構造をつなげば、それはこの法線方向に対するエネルギー障壁を与える。

一方で、force inversion 法ではポテンシャルエネルギー曲面上で、エネルギー障壁の方向をよく近似する超空間の法線ベクトルを用意し、構造最適化で用いる力場のこの法線に対して平行な成分を反転してやることで、元々の力場のヘッセ行列が鞍点をとる地点をヘッセ行列が極小値をとる地点に変換する。そして後は通常の構造最適化をこの力場について行えばポテンシャルエネルギー曲面の鞍点が求まる。良くない法線ベクトルや鞍点から遠い初期構造に対しては計算が不安定になるのである程度 hyper plane constraint 法などで鞍点の場所と初期構造を絞り込まないと使えない。また、ポテンシャルエネルギー曲面が鞍点でなめらかな構造をもたない場合、例えば障壁の上で電子状態が交差する場合にはこの手法は不安定で使えない。

なお、この二つの手法のためのツールとして、論理機番 99 に出力される二つのサマリファイル中の構造データから差分ベクトルを作るツール diffstr と、二つのサマリファイルを構造を線形に補間した構造データを作るツール iplstr が付属している。

また、現状の hyper plane constraint 法と force inversion 法の実装ではセル構造の変形はサポートされていないので注意すること。

6.3.1 入出力

cgmrpt は論理機番 10 から入力データを読み込む。MPI 並列実行している場合すべての MPI プロセスがこの入力データを読み込む。

cgmrpt は inipot が作成したデータを必要とし、論理機番 13,14,15,40,22,73,74 から読み込む。これら読み込みは rank 0 が担当する。21 から読み込むがこれは後方互換性のためである。

cgmrpt は初期ローカルポテンシャルと初期波動関数を論理機番 11,95 からそれぞれ読み込むことができる。これら読み込みは rank 0 が担当する。なお平面波基底のカットオフや波動関数のサンプル k 点が異なっても読み込むことができる。また初期ローカルポテンシャルとして、イオンのローカルポテンシャルか、読み込んである原子の電荷分布の重ね合わせから計算したローカルポテンシャルを使うこともできる。読み込みを行わない場合初期波動関数はランダムに生成される。

構造最適化時の原子構造、全エネルギーなどの情報は rank 0 が標準出力のログに出力する。

cgmrpt は計算終了時にローカルポテンシャルと電荷分布を論理機番 11, 25 にそれぞれ出力する。また論理機番 96 に波動関数を書き出すことができる。このほか論理機番 99 に構造最適化した原子構造、全エネルギー、原子に働く力、ストレス、スピン偏極度など結果の要約をまとめたものを出力する。これら書き出しは rank 0 が担当する。

出力された電荷分布、ローカルポテンシャル、波動関数を OpenDX で読み込める形式に変換するプログラムはそれぞれ rho2dx, lpt2dx, wfn2dx として xTAPP-util に含まれている。

構造最適化の計算の継続を行うために必要なデータは、論理機番 70,71,72 に rank 0 から随時書き出されている。計算が時間切れとなって終了した場合、次の計算実行時に継続を指示すると、これらデータとローカルポテンシャル、波動関数から構造最適化が継続される。また、構造最適化が収束したとき論理機番 70 にはこの結果が書き込まれている。

この他、cgmrpt は 51 に固有エネルギーの \cos 展開データを記録する。この記録は pefcos のためのものである。

6.3.2 論理機番 99 のフォーマット

```
&struct_data
lattice_factor = ...
lattice_list = ...
total_energy = ...
stress_tensor = ...
fermi_energy = ...
number_element = ...
number_atom = ...
spin_polarization = ...
spin_polarization_vector = ...
abs_spin_polarization = ...
abs_spin_polarization_vector = ...
/

# valence_charge, nucleous_charge
zo  zn
...

# atom_kind, atom_position by cell coordinate
atom_kind, pos_a, pos_b, pos_c
...

# force by Cartesian coordinate
```



```

force_x force_y force_z
...

# orbital eigen value
orbital eigen value informations for each sampled k points
...

```

となっている。入力ファイルフォーマットと同じキーワードは同じ意味を持つ。その他のキーワードは

total_energy 全エネルギー E

stress_tensor ストレステンソル

$$-\frac{\partial E}{\partial \epsilon_\nu}; \quad \nu \text{は } xx, yy, zz, yz, zx, xy \text{ の順}$$

fermi_energy フェルミエネルギー。それぞれのスピン成分ごとに個数を決めるモードがあるので二つある。

spin_polarization スピン偏極

$$\int_{\Omega} (\rho_2(\mathbf{r}) - \rho_1(\mathbf{r})) d\mathbf{r}$$

ただし non-collinear 磁性の計算をしているときは

$$\int_{\Omega} m_z(\mathbf{r}) d\mathbf{r}$$

spin_polarization_vector ベクトルとしてのスピン偏極の Cartesian 座標での 3 成分

$$\int_{\Omega} m_x(\mathbf{r}) d\mathbf{r}, \int_{\Omega} m_y(\mathbf{r}) d\mathbf{r}, \int_{\Omega} m_z(\mathbf{r}) d\mathbf{r}$$

non-colliner 磁性の計算をしているときのみ出力する。

abs_spin_polarization 絶対スピン偏極 (アンチフェロの時に値が 0 にならないもの。)

$$\int_{\Omega} |\rho_2(\mathbf{r}) - \rho_1(\mathbf{r})| d\mathbf{r}$$

ただし non-collinear 磁性の計算をしているときは

$$\int_{\Omega} |\mathbf{m}(\mathbf{r})| d\mathbf{r}$$

abs_spin_polarization_vector ベクトルとしてのスピン偏極の Cartesian 座標での各成分の絶対値の積分

$$\int_{\Omega} |m_x(\mathbf{r})| d\mathbf{r}, \int_{\Omega} |m_y(\mathbf{r})| d\mathbf{r}, \int_{\Omega} |m_z(\mathbf{r})| d\mathbf{r}$$

non-colliner 磁性の計算をしているときのみ出力する。spin_polarization_vector が 0 でもこの値が 0 でないということは、偏極が空間変動しているが総和は打ち消しあって 0 になっていることを示している。

force_x, force_y, force_z 原子に働く力 (デカルト座標)

である。

6.3.3 replica 実行

計算機システムによっては、入力パラメータを多数用意してそれを並列実行させることが困難なものがある。このようなシステムで複数入力パラメータでの並列実行を行うために replica 実行機能が用意されている。レプリカ実行を行うためには通常の入力データを複数用意し、cgmrpt 自体にはそれらを複数並列実行するための特別な入力を与える。

入力の書式は

```
REPLICA
color(1) key(1)
color(2) key(2)
...
color(nproc) key(nproc)
dirprefix ifname ofname
```

となっている。ここで color(i) と key(i) は、MPI_COMM_WORLD で rank=i の MPI process を所属させる新しいコミュニケータのカラー番号と新しいコミュニケータでの rank を決めるためのキーである。詳しくは MPI_COMM_SPLIT のマニュアルを参照すること。同一のカラーの rank の MPI process は同一の新しいコミュニケータを構成し、その中で一つの入力ファイルが実行される。実行される入力ファイルは、cgmrpt が実行されているディレクトリ以下に、dirprefix に続けてカラー番号を先頭に 0 を補って 5 桁に揃えた数字をつなげた名前のディレクトリの中の ifname の名前のファイルである。そしてプログラムの出力はこのディレクトリの ofname の名前のファイルに格納される。

なお、これらのディレクトリは inipot が生成するデータファイルをそれぞれ含んでいなければならない。inipot も replica 実行できるので、replica 実行でそれらを作成させるのが良い。

例えば、

```
REPLICA
0 0
1 0
p. si.cg cgmrpt.log
```

の場合、rank 0 と rank 1 の MPI process がそれぞれディレクトリ p.00000 と p.00001 でその中の si.cg を入力ファイルとして実行される。出力はこれらディレクトリの中の cgmrpt.log に出力される。

- [1] Donald G. Anderson, J. Assoc. Computing Machinery, 12, 547 (1965)
- [2] M. Methfessel and A.T. Paxton, PRB 40 (1989) 3616
- [3] G. Kresse and J. Furthmueller, Computational Materials Science 6 (1996) 15
- [4] M. Bernasconi et al., J. Phys. Chem. Solids **36** 501-505 (1995).
- [5] Y. Tateyama, T. Ogitu, K. Kusakabe, and S. Tsuneyuki, PRB 54, 14994 (1996).

6.4 vbpef と pefocs

vbpef は cgmrpt で計算したローカルポテンシャルと波動関数 (hybrid 汎関数の場合) で決まるハミルトニアンを固定して、逆空間上に設定する k 点の軌跡の上で、固有エネルギー、波動関数、軌道電荷分布の空間積分値、軌道電荷分布そのものを求めるプログラムである。いわゆるバンド図のデータを生成するのはこのプログラムである。

一方で pefcos はバンド図のデータだけを cgmprpt で計算した固有エネルギーの cos 展開から計算するプログラムである。そのため cos 展開の対象となったバンドの範囲だけ計算できる。

k 点の軌跡は複数区間のつながった線分で設定する。また軌道電荷分布の積分は指示した点を中心とする球状、またはセルの軸に平行になる平板状のいずれかでできる。

計算したバンドの固有エネルギーデータは論理機番 50 に出力される。これを xTAPP-util の vbpef2gp-ksda で処理するとバンド図を描ける。

計算した軌道電荷の積分データは論理機番 51 に出力される。このファイルのフォーマットはこの文書のファイルフォーマットの節にある。計算した k 点の波動関数データは 58, 計算した軌道電荷データは 57 にそれぞれ出力される。xTAPP-util 中の wfk2dx および rok2dx で 58 および 57 を OpenDX 向きのデータ形式に変更できる。

vbpef は inipot が作成したデータを必要とし、論理機番 15,40,22,73,74 から読み込む。これら読み込みは rank 0 が担当する。

vbpef に cgmprpt で構造最適化した結果を引き継ぎたい場合、chain_calc=3 とセットする。論理機番 70 から cgmprpt が構造最適化した結果が読み込まれる。

TAPP にあった vbwfn の機能は vbpef に統合されているので、必要ならこちらを使うこと。

6.5 vbstm

cgmprpt で計算したローカルポテンシャルと波動関数 (hybrid 汎関数の場合) から STM 像のシミュレーションを局所状態密度をフェルミエネルギーからバイアス電圧まで積分して行うプログラムである。つまり Tersoff Hamann 近似 [1] での計算を行なう。

フェルミエネルギーは、入力として与えなければならない。

計算した STM 像は論理機番 80 に出力される。この出力ファイルを OpenDX で読み込める形式に変換するプログラム stm2dx が xTAPP-util に入っている。

mdrpt は inipot が作成したデータを必要とし、論理機番 13,14,15,40,22,73,74 から読み込む。これら読み込みは rank 0 が担当する。21 から読み込むがこれは後方互換性のためである。

vbpef に cgmprpt で構造最適化した結果を引き継ぎたい場合、chain_calc=3 とセットする。論理機番 70 から cgmprpt が構造最適化した結果が読み込まれる。

[1] J. Tersoff and D.R. Hamann, Phys. Rev. Lett. **50** 1998–2001 (1983); Phys. Rev. B **31** 805–813 (1985)

6.6 wfn2chg

cgmprpt で計算した波動関数から、電荷分布の空間積分を固有エネルギーで分解したものや projected DOS を計算するプログラムである。

このプログラムは実際に MPI 並列動作するものではないが mpirun で起動する必要がある。

電荷分布の空間積分は指示した点を中心とする球状、またはセルの軸に平行になる平板状のいずれかでできる。

projected DOS の計算は、原子の動径波動関数に対するものと最局在 Wannier 関数に対するものの 2 種類を計算できる。

原子の動径波動関数は論理機番 18 から読み込む。このファイルは

```
nsmp1 wavil
r(1) wav(1)
r(2) wav(2)
```

```
...
r(nsmpl) wav(nsmpl)
```

の形式のもので、主量子数 n 、軌道角運動量 l の動径波動関数を $R(n, l; r)$ とするとき、1 から $nsmpl$ までのログメッシュ $r(1:nsmpl)=r$ 上の角運動量 $wavil = l+1$ 、の動径波動関数 $wav(1:nsmpl)=rR(n, l; r)$ を指定するものである。これは擬ポテンシャルで原子を解くプログラム (solps) で作ることができる。

この動径波動関数を置く位置は、計算上原子がある位置を原子の番号で指定するか置く位置をセル座標で指定するかで決める。ただし、計算上原子のない位置に指定する場合でかつ読む込む動径波動関数が Ultrasoft 擬ポテンシャルによるものでノルムが保存していない場合、得られる projected DOS の全エネルギーでの積分は 1 にならないので解釈に注意が必要である。例えば電荷の量として解釈することはできない。環境の違いによる値の比には意味があるであろう。

最局在 Wannier 関数に対する projected DOS を計算する場合、対象の電子状態を計算する k 点メッシュと最局在 Wannier 関数を計算する k 点メッシュは同一でなければならないが、両者を計算する原子構造そのものが同一である必要性はない。

計算した積分値や projected DOS は論理機番 55 に出力される。このデータを処理してエネルギーの関数としてのグラフを描くプログラムが xTAPP-util にあり、積分値用が wfchg2pdos、projected DOS 用が ltzpdos である。これらのプログラムは今のところ、これらに与えるローレンチアンとの畳み込みによってエネルギー依存性の計算を行っている。なお、ltzpdos の出力には total DOS も含まれている。

projected DOS の計算を行なっている場合には、計算に用いている k 点メッシュが Γ 点を含む単一のメッシュである場合、論理機番 55 に加えて論理機番 59 に四面体法による projected DOS の計算を行なうためのデータが格納される。このデータはこの文書で説明されているツール、tetrapdos で処理することでエネルギーの関数としてのグラフのデータに変換できる。なお、tetrapdos の出力は ltzpdos と互換性がある。

原子軌道への projected DOS の角運動量は $wavil = 3(l = 2; d \text{ 軌道})$ まで指定可能である。また出力における磁気角運動量の並び順はカーテシアン座標を用いて、 p 軌道の場合

$$y, z, x$$

d 軌道の場合、

$$xy, yz, 3z^2 - 1, zx, x^2 - y^2$$

の順となっている。

また、cgmrpt で計算した k 点の軌道電荷を出力させることも出来、これは論理機番 57 に出力される。このファイルは vbpef と同様に xTAPP-util の rok2dx で OpenDX 用のフォーマットに変換できる。

wfh2chg は inipot が作成したデータを必要とし、論理機番 13,14,15,40,22,73,74 から読み込む。21 から読み込むがこれは後方互換性のためである。

6.7 mdrpt

mdrpt は第一原理分子動力学 (BOMD) を行うプログラムである。mdrpt は cgmrpt と同様の電子状態計算を行って原子に働く力を計算する。

速度ベルレ法による NVE アンサンブル、速度ベルレ法と Berendsen thermostat の組み合わせによる NVT アンサンブル、および Stern のアルゴリズム [1] による NPT アンサンブルが扱える。

なお、この NPT アンサンブルでは `barostat` が止まるパラメータを設定することで NVT アンサンブルにすることもできる。

分子力学の原子位置などの記録は、論理機番 16 にバイナリで出力される。出力は rank 0 が行う。このファイルを読み取ってアスキー形式で出力するプログラム `scanmdlog` が `xTAPP-util` に入っている。

初速度は設定することもできるが、設定温度に合うようなランダムな初速を与えることもできる。

MD ステップ毎に波動関数とローカルポテンシャルを次のステップに補外させて収束を加速することができる。[2,3]

`mdrpt` は `inipot` が作成したデータを必要とし、論理機番 13,14,15,40,22,73,74 から読み込む。これら読み込みは rank 0 が担当する。21 から読み込むがこれは後方互換性のためである。

`mdrpt` は初期ローカルポテンシャルと初期波動関数を論理機番 11,95 からそれぞれ読み込むことができる。これら読み込みは rank 0 が担当する。なお平面波基底のカットオフや波動関数のサンプル k 点が異なっても読み込むことができる。また初期ローカルポテンシャルとして、イオンのローカルポテンシャルか、読み込んである原子の電荷分布の重ね合わせから計算したローカルポテンシャルを使うこともできる。読み込みを行わない場合初期波動関数はランダムに生成される。

`mdrpt` は計算終了時にローカルポテンシャルと電荷分布を論理機番 11, 25 にそれぞれ出力する。また論理機番 96 に波動関数を書き出すことができる。これら書き出しは rank 0 が担当する。

分子力学が設定ステップまで進まずに終了した場合、論理機番 75 に継続実行のためのデータが rank 0 によって書き出される。次の実行で継続を指示するとこのデータが rank 0 から読み取られて継続実行が始まる。この際、ローカルポテンシャルと波動関数の読み込みも同時に必要である。

[1] H.A. Stern, J. Compt. Chem, Vol. 25, No. 5, 749-761 (2004)

[2] T.A. Arias, M.C. Payne, J.D. Joannopoulos, Phys. Rev. B 45 (1992) 1538

[3] Dario Alfe, Comp. Phys. Comm., 118 (1999) 31

6.8 wannier

`cgmrrpt` で計算した波動関数から、最局在ワニエ関数 [1,2] を計算するプログラムである。

このプログラムは実際に MPI 並列動作するものではないが `mpirun` で起動する必要はある。

このプログラムには二つのモードがある。一つは逆空間でのワニエ関数の計算を行なうモード `generate_rcp`、もう一つは計算済みの逆空間でのワニエ関数の実空間表現を計算するモード `recp_to_real` である。

このプログラムを使う場合、計算セルの形状のとり方に制限ができるので注意すること。またその形状がなにであるのかを別途指定する必要もある。他に、計算に使用したサンプル k 点メッシュは Γ 点を含み、対称性を満たす単一の格子でなければならない。

`generate_rcp` モードでは、論理機番 113 に逆空間表現のワニエ関数データ、論理機番 150 にワニエ関数をプロット関数から構成する際の変換行列、論理機番 122 にワニエ関数によってハミルトニアンを計算したもの、論理機番 134 にワニエ関数の実空間での中心位置がカーテシアン座標で、それぞれ書き込まれる。

`recp_to_real` モードでは、論理機番 113 からあらかじめ計算した逆空間表現のワニエ関数データが読み込まれ、論理機番 116 にワニエ関数の実空間表現のデータが書き込まれる。

論理機番 113 には j 番めのワニエ関数 $w_j(\mathbf{r})$ を

$$w_j(\mathbf{r}) = \frac{1}{N_k \sqrt{\Omega}} \sum_{\mathbf{k}, \mathbf{G}} \exp(i\mathbf{k} \cdot \mathbf{r}) w_{\mathbf{k},j}(\mathbf{G}) \exp(i\mathbf{G} \cdot \mathbf{r})$$

と構成するときの $w_{\mathbf{k},j}(\mathbf{G})$ が格納されている。この規格化は、

$$\sum_{\mathbf{G}} |w_{\mathbf{k},j}(\mathbf{G})|^2 = 1$$

となっている。これは

$$\int d\mathbf{r} |w_j(\mathbf{r})|^2 = 1$$

に対応している。

論理機番 150 にはこのワニエ関数 $w_{\mathbf{k},j}(\mathbf{G})$ をブロッホ関数 $\phi_{\mathbf{k},n}(\mathbf{G})$ から構成する際の変換行列 $A_{n,j}(\mathbf{k})$ が格納される。ここで

$$w_{\mathbf{k},j}(\mathbf{G}) = \sum_n \phi_{\mathbf{k},n}(\mathbf{G}) A_{n,j}(\mathbf{k})$$

である。格納時の添字の順番は内側から j, n, \mathbf{k} となっている。また可約 \mathbf{k} 点すべてについて出力されている。出力する \mathbf{k} 点の順は標準出力に出てくる可約 \mathbf{k} 点のリストの順である。

論理機番 116 へのワニエ関数 $w_j(\mathbf{r})$ の \mathbf{r} の範囲は、 a, b, c 軸単位で -1 から 1 までである。ファイルの書式は xTAPP の電荷密度ファイルと同一である。(スピン成分は一つ)。可視化する際には xTAPP-util の rho2dx ないし rho2xsf を用いること。

ワニエ関数によって内挿されたバンド分散を計算する場合、付属のツール hmatr2bnd を用いる。使用方法はほとんど pefcos や vbpef と同じであるが論理機番 122 に出力されているハミルトニアンを使う点が異なる。また計算の対象となるバンドはワニエ関数を計算したバンドに固定され、バンドの指定はできない。

自発分極の計算のため、分極のうちイオンによる成分 Ionic polarization と、求めたワニエ関数の中心位置の総和 Sum of the Wannier center がそれぞれデカルト座標で標準出力に出力されている。すべての占有状態についてワニエ関数を計算している場合、ワニエ関数の中心位置の総和は分極のうち計算しているスピン方向の電子による成分となる。Sum of the Wannier center は二種類 (optimized と invariant) に分かれているが、optimized は [1] で定義される最局在ワニエ関数の計算に伴う定義によるもの、invariant はワニエ関数のゲージに依存しないものである [3]。invariant, aligned とあるものは格子ベクトル分の不定性を optimized にほぼ一致するように補正してあるものである。最局在ワニエ関数を収束させなくても同一の値になる点で、分極の計算には invariant を使うのが良い。なお、うまくワニエ関数の初期値がとってある場合 Ionic polarization と optimized から単純な引き算で分極が計算できるはずであるので、これにそろっている invariant, optimized が最適な選択のはずである。

このプログラムはウルトラソフト擬ポテンシャルに対応するが、その方式は参考文献 [4] によっている。

- [1] N. Marzari and D. Vanderbilt, Phys. Rev. B 56, 12847 (1997)
- [2] I. Souza, N. Marzari, and D. Vanderbilt, Phys. Rev. B 65, 035109 (2001)
- [3] R. D. King-Smith and D. Vanderbilt, Phys. Rev. B 47, 1651 (1993)
- [4] A. Ferretti, A. Calzolari, B. Bonferroni and R. Di Felice, J. Phys: Condens. Matter 19, 036215 (2007)

6.9 pe2dos

pe2dos は状態密度を cgmrrpt で計算した固有エネルギーの cos 展開から計算するプログラムである。そのため cos 展開の対象となったバンドの範囲だけが計算の対象となる。

```
$ pe2dos fname ndx ndy ndz nsmp1 mode
```

で起動する。ここで fname は cgmrrpt が出力した論理機番 51 のファイルである。ndx, ndy, ndz はブリュアンゾーンのメッシュ生成に用いる分割数であり、a,b,c 軸がそれぞれこの 2 倍で分割される。状態密度はメッシュ上のエネルギーの最小値から最大値までを nsmp1 等分したサンプル点で計算される。mode には 0 または 1 を指定し、0 の場合、出力はそのエネルギーまでの状態数（つまり状態密度の積分値）となり、1 の場合、出力はそのエネルギーでの状態密度となる。

標準出力に出力される結果はすべて原子単位系を用いている。

6.10 tetrapdos

tetrapdos は projected DOS を wfn2chg が論理機番 59 に出力するデータから四面体法を用いて計算するプログラムである。wfn2chg はサンプル k 点が単一の Gamma 点を含む格子をなしているときにのみ、59 にこのデータを出力する。

```
$ tetrapdos fname nsmp1 mode
```

で起動する。ここで fname は wfn2chg が出力した論理機番 59 のファイルである。projected DOS は計算の対象となった軌道エネルギーの最小値から最大値までを nsmp1 等分したサンプル点で計算される。mode には 0 または 1 を指定し、0 の場合、出力はそのエネルギーまでの状態数（つまり状態密度の積分値）となり、1 の場合、出力はそのエネルギーでの projected DOS となる。

標準出力に出力される結果はすべて原子単位系を用いている。この出力の書式は xTAPP-util の ltzpdos と同様である。出力データの行のフォーマットは、LDA なら

```
eng, tdos, (pdos(im),im=1,nwav)
```

LSDA なら

```
eng, tdos(1), (pdos(im,1),im=1,nwav), (pdos(im,2),im=1,nwav)
```

である。ここで nwav は計算の対象となった軌道の数、eng はエネルギー、tdos は total DOS、pdos は各軌道ごとの projected DOS。(1),(2) はスピンの番号。pdos, tdos とともに [states/a.u.] である。

原子軌道に対する projected DOS の場合、nwav = 2*wavil-1 である。この場合、各軌道はカーテシアン座標で実数化された磁気角運動量に対応するものであり、その並びは wavil=2 (p) なら、y,z,x 成分であり、wavil=3 (d) なら、xy,yz,3*z²-1,zx,x²-y² 成分である。

一方で、Wannier 関数に対する projected DOS の場合、nwav は計算した Wannier 関数の数である。

6.11 xticonv

xticonv は入力ファイルの中の原子構造データを可視化ツールのために変換するプログラムである。対応フォーマットは CIF(<http://www.iucr.org/resources/cif>) と XYZ である。プログラムは

```
$ xticonv [fmt] [file name]
```

として起動する。ここで [fmt] は

cif: CIF 形式

xyz: XYZ 形式

xyzpr: XYZ 形式であるが、セルの境界付近 (セルの各周期の方向に 0.1 a.u. まで) に位置する原子を重複して出力

xyzmol: XYZ 形式であるが、セルの周期性を使って 1 番目の原子にできるだけ他の原子の位置を近づけて出力。孤立分子系の出力用。

poscar: VASP の POSCAR 形式。第一行には使用している原子種が書き込まれる。

であり、[file name] は変換したい入力ファイル名である。変換結果は標準出力に出てくる。

6.12 strconv

strconv は cgmrrpt が論理機番 99 に出力する構造最適化した原子構造を可視化ツールのために変換するプログラムである。対応フォーマットは CIF(<http://www.iucr.org/resources/cif>) と XYZ である。プログラムは

```
$ strconv [fmt] [file name]
```

として起動する。ここで [fmt] は

cif: CIF 形式

xyz: XYZ 形式

xyzpr: XYZ 形式であるが、セルの境界付近 (セルの各周期の方向に 0.1 a.u. まで) に位置する原子を重複して出力

xyzmol: XYZ 形式であるが、セルの周期性を使って 1 番目の原子にできるだけ他の原子の位置を近づけて出力。孤立分子系の出力用。

poscar: VASP の POSCAR 形式。第一行には使用している原子種が書き込まれる。

cforce: 原子に働く力を入力ファイルでの原子の並び順にデカルト座標、原子単位系で出力する。phonopy と組み合わせるときなどに使う。

であり、[file name] は変換したい入力ファイル名である。変換結果は標準出力に出てくる。

6.13 diffstr

diffstr は cgmrrpt が論理機番 99 に出力する構造最適化した原子構造を二つ受け取って、その構造データの差分ベクトルをセル座標で出力するツールである。差分ベクトルは周期性を考慮して、セル座標の差分の絶対値ノルムが最小になるようにとられる。

```
$ diffstr [file name 1] [file name 2]
```

として起動する。差分は $2 - 1$ として計算される。結果は標準出力に出てくる。

6.14 iplstr

iplstr は cgmprpt が論理機番 99 に出力する構造最適化した原子構造を二つ受け取って、その構造データを線形に補間した構造データをセル座標で出力するツールである。

```
$ iplstr [file name 1] [file name 2] tipl
```

として起動する。補間は $1 + tipl * (2 - 1)$ として計算される。なお、差分の部分は周期性を考慮して、セル座標の差分の絶対値ノルムが最小になるようにとられる。結果は標準出力に出てくる。

6.15 cmpstr

cmpstr は cgmprpt が論理機番 99 に出力する構造最適化した原子構造を二つ受け取って、それらが誤差の範囲で同一であるかチェックするツールである。これは、プログラム開発時の動作チェックのために使用する。

```
$ cmpstr [file name 1] [file name 2]
```

として起動すると、変化があれば終了状態が 0 でなくなり標準出力に変化の内容について出力する。

```
$ cmpstr
```

として起動すると、誤差として認める範囲を標準出力に出力する。

7 ESM 機能

大谷、杉野による有効遮蔽体法 (ESM 法) を使用することができる。

この機能を利用する場合、標準の入力に加えて ESM のための入力ファイル esmprm.dat を用意する。このファイルがプログラムのカレントディレクトリに存在することで ESM 機能が有効になる。このファイルはすべての MPI process から読み込まれる。

ESM 機能を使う場合、計算セルは ESM を適用している軸が他の軸に直交するように取らなければならない。

esmprm.dat については、inputformat.tex を参照せよ。

8 擬ポテンシャル

TAPP-1.0.8 で用いられていた形式の擬ポテンシャルのフォーマットを用いるがヘッダは異なっており、

```
#xPSV-1
# ATOM NAME: {atom_name}
# {comment}
# ...
xcname
zo zn
...
```

で始まるようになっている。ここで{atom_name}は原子名を表す文字列、{comment}はその他のコメントである。{atom_name}はログ中で出力される。xcname は交換相関ポテンシャルのタイプを表す。zo はこの擬ポテンシャルのイオンの価数であり、zo はこの擬ポテンシャルの原子番号である。なお、zn は省略可能である。元々のフォーマットでは zn は存在していない。

TAPP-3.0 系列のフォーマットと TAPP-1.0.8 のフォーマットでは initial spin のデータが増設された違いがある。このデータは xTAPP では使えないので削除する必要がある。

9 原子の電荷分布

28 から始まる論理機番から入力する原子電荷分布は

```
nsmpl
r(1) rho1(1) rho2(1)
r(2) rho1(2) rho2(2)
...
r(nsmpl) rho1(nsmpl) rho2(nsmpl)
```

の書式で与える。ここで nsmpl は動径のログメッシュの数であり、r はログメッシュ、rho1、rho2 は電荷の動径分布である。rho2 がない場合にはスピン偏極なし、ある場合にはスピン偏極ありとして扱われそれぞれ成分 1、2 となる。このデータは solps か solspin を用いて計算することができる。

10 ログの見かたについて

10.1 実行条件の確認

ログの先頭部分には入力ファイル形式と同じ形式で BEGIN INPUT から END INPUT まで実行に使われたパラメータが出力されている。入力で省略したものにプログラムが当てはめた値もここで確認できる。

10.2 SCF の収束の確認

SCF の収束はログの SQUARE ERROR = で確認する。この定義はデフォルトでは

$$\sum_{\sigma} \frac{1}{\Omega} \int_{\Omega} |V_{\sigma, new}(r) - V_{\sigma, old}(r)|^2 dr = \sum_{\sigma, G} \left(\frac{1}{\Omega} |V_{\sigma, new}(G) - V_{\sigma, old}(G)| \right)^2$$

である。この定義は収束度がシステムサイズに依存しにくくなるので良い。

しかし、TAPP-3.0 との互換性を指定している場合は、これをさらに 2*nvxyz で割ったものが定義となる。ここで nvxyz はローカルポテンシャルの逆空間での全メッシュ数である。

なお、この収束の定義では非占有状態の収束の度合いを知ることは出来ない。このためののちの計算の都合で非占有状態を収束させたいときには別途 IKS, EIGEN, RESID, SIN, EPS を調べる必要がある。これは SCF 一回につき、一塊のログが k 点の数だけ出るものであるが、EIGEN が一回前の SCF との固有値の変化の絶対値を表している。これが十分に小さくなるようにしなければならない。なお、非占有状態を収束させるためには SCF を回すことなく対角化ルーチンを回すことが必要である。そのため非占有状態を特に収束させたいときには SQUARE ERROR =

が十分に小さくなった状態で davidson_number_diag を大きめに取って対角化を十分に行わせれば良い。

なお、number_density = 4 で計算している場合、すなわちノンコリニアで計算している場合、SQUARE ERROR の定義は

$$\sum_{\nu=0}^3 \frac{1}{\Omega} \int_{\Omega} |V_{\nu}(r) - V_{\nu,old}(r)|^2 dr = \sum_{\nu,G} \left(\frac{1}{\Omega} |V_{\nu,new}(G) - V_{\nu,old}(G)| \right)^2$$

と拡張される。ここで $\nu = 0$ は電荷密度、 $\nu = 1, 2, 3$ はスピン密度に対応する。

10.3 全エネルギー

SCF の収束の過程で出力される ETEIG はバンドエネルギーから簡易に計算した変分量になっていない全エネルギーである。ただし hybrid 計算の場合にはこの計算が低コストでできないため正しく行っていない。ETEIG は力の計算が終わった後に出力される変分量になっている全エネルギー”TOTAL ENERGY”に近い値になるものである。

10.4 構造最適化と力の残差

構造最適化の過程で収束を見るには MAX FORCE を調べれば良い。これは原子に働く力のうち最大のもの [hartree/bohr] である。セル形状を最適化している場合、これに加えて MAX STRESS を調べる。これは、

$$\frac{1}{N_{atom}} \left(\frac{dE}{d\epsilon_{\nu}} - p_{ext} V \right); \quad \text{for } \nu = xx, yy, zz$$

$$\frac{1}{N_{atom}} \left(\frac{dE}{d\epsilon_{\nu}} \right); \quad \text{for } \nu = yz, zx, xy$$

の最大である。

10.5 実行プロファイルの確認

SCF が終了する時と、力の計算が終了するときにそこまでの実行プロファイルが profile: time count に続いて出力される。最初の数字が経過時間の合計、次の数字が実行回数である。

10.6 全サンプル k 点数とサンプル k 点の位置の確認

inipot が全サンプル k 点を決定している。このログのうち NI= で現れているのが全サンプル k 点の数である。なおこの数は設定した k 点メッシュより多くなることがある。これは設定した k 点メッシュに対称性でつながっている k 点も数えるためである。また、この NI= の直後にサンプル k 点の番号とその位置の対応が一行に 2 セットずつ出ている。1 セットの数字のうちもっとも右側のものは、サンプルした k 点が対称操作を有効活用する割合を示しており 1 が最大である。位置は逆格子の基本ベクトル単位で表示されている。

10.7 平面波基底の数の確認

inipot のログにサンプル k 点の番号ごとの平面波基底の数が iks, nkm = で記録されている。

10.8 各種メッシュ

電荷分布: inipot のログに NRX, NRY, NRZ で現れている。それぞれの 2 倍が実際に使われるメッシュ数である。NRXYZ は総メッシュ数である。

波動関数の実空間メッシュ: inipot のログに nwx, nwy, nwz で現れている。それぞれの 2 倍が実際に使われるメッシュ数である。

11 プログラムの運用について

11.1 セルの取り方

フーリエ変換メッシュとセルの対称性が矛盾しない必要がある。このため、セルの a, b, c 軸は対称性よくとる必要があり、そうでない場合には実行初期にエラーになる。body centered tetragonal のケースではセルの取り方が悪いと実際にエラーになる。

11.2 原子のスピン偏極の初期条件の設定

antiferro となる系の場合など、初期スピン電荷密度を目的の秩序状態に初期化したくなる場合がある。この目的で、ある原子の初期電荷密度を上向きと下向きの反対に偏極した二種類にわけて設定したい場合、対応は二つある。

1. # initial charge セクションを使う。このセクションで初期スピン電荷の偏極の向きを原子ごとに反転できる。詳細は inputformat.tex を見ること。

原子のスピン電荷密度を必要に応じて反転しながら初期電荷をつくることができる。

2. これらスピンの極性が異なる原子に入力上、異なる原子種の番号を割り当てたうえで擬ポテンシャル本体は共通、28 から始まる論理機番から入力する原子の電荷分布データはそれぞれ反対向きに偏極したものとすれば良い。

いずれの場合でも、原子の電荷分布データから初期ローカルポテンシャルを生成させるために入力ファイルの namelist &tappinput において initial_lpt を 2 にセットすることを忘れないこと。

11.3 スピン軌道相互作用、non-collinear 磁性の計算

namelist tappinput において number_component = 2 と設定することでスピン軌道相互作用の計算機能が有効になる。この際スピン軌道相互作用を含む擬ポテンシャルを使う必要がある。これを指定するとプログラムは 2 成分波動関数 (Pauli 方程式) を扱うようになる。

入力で指定するバンドの本数は各バンドにそれぞれ 1 個の電子が収容されるものとして与えること。つまり通常に比べて 2 倍のバンド数が必要になる。この様にバンドの本数が 2 倍必要となり波動関数の基底の数が 2 倍になるので、必要計算量は通常の 8 倍になり、必要メモリ量は通常の 4 倍になる。

またデフォルトでは時間反転対称性が仮定されて磁性は現れない。non-collinear 磁性の計算をするには加えて number_density = 4 と設定する。このように設定すると時間反転対称性が仮定されなくなり、ゼロではないスピンベクトル密度での計算ができるようになる。

一般化勾配近似 (GGA) を交換相関汎関数に用いている場合、number_density = 4 では SCF が収束しないことがある。この場合で計算結果が実は collinear になるケースの場合、noncol_spin_axis をこの分極方向に設定すると収束することがある。

non-collinear 計算の場合、初期スピン電荷密度を指定して計算を始める必要があるが、このためには# initial charge セクションを使う。このセクションの namelist initial_charge に mode = 'direct_atomic_charge' を指定すると、スピン分極の向きを含めて初期電荷密度の設定ができる。詳細は inputformat.tex を見る。また、この初期電荷密度から初期ローカルポテンシャルを生成させることを忘れないこと。対称性データはこの初期電荷密度と無矛盾でなければならないが、スピン偏極が存在する場合の可能な対象操作の列挙にも FINDSYM (<http://stokes.byu.edu/iso/findsym.php/>) が利用できる。

各バンドあたり各原子につくるスピン分極の向きを計算したい場合には、vbpef の入力に知りたい原子位置に密度計算の球を設定してから vbpef に distrib_mode='sphere' を用いて計算させ、出力の論理基盤 51 を見れば良い。

11.4 計算条件の改良

同じ計算条件ではない波動関数、ローカルポテンシャルを読むことが出るので計算条件の逐次改良が可能である。ただし、2 成分波動関数 (Pauli 方程式) と 1 成分波動関数 (Schrödinger 方程式) の間の相互変換はまだサポートされていない。

11.5 メモリ

最近のハードウェアはメモリが多く搭載されているため、波動関数などのデータはすべてメモリ上に載せられている。オリジナルの TAPP のように途中でディスクに置くことはできない。

11.6 ファイル I/O

通常 Fortran の処理系は環境変数をセットすることであらかじめ論理機番を特定のファイルに結びつけておくことができる。この機能を用いて入出力をセットアップするスクリプトを用意してその中から各プログラムを呼び出すようにするのが良いだろう。

ただし、gfortran(version 4.7.2) ではこの機能は存在しないようである。そのために、この機能を代替するコードが xTAPP には含まれている。ただし、同一のファイルを異なる論理機番に対応させることができないため、波動関数の入出力においては注意が必要である。また、この機能を使わない場合 fort.* のファイルがそのまま各論理機番に対応するので、あらかじめ fort.* を目的のファイルへのシンボリックリンクとすることもできる。

このような fortran の機能を使う方式の他、オプションのセクションである file map data セクションでファイル名をプログラムに直接与えることもできる。ただし、この場合でも最初の入力となる論理機番 10 だけは環境変数なしシンボリックリンクで与えなければならない。

また MPI 並列実行している場合、ログを標準出力に出力するのは rank 0 だけである。その他の I/O も基本的に rank 0 が担当するようにしてある。

11.7 MPI と環境変数と論理機番

論理機番を環境変数を使ってプログラムに渡す場合、MPI の実装によっては mpirun の引数の中に記述するなど特定の方法を取らない限りプログラムに環境変数を伝達できないことがあるので、注意が必要である。

11.8 波動関数の分散

波動関数の分散は平面波基底で行っており、FFT が必要になるときだけデータをバンド分散に持ち替えている。ローカルポテンシャル、電荷分布などは各 MPI プロセスがコピーを持っている。

11.9 hybrid 並列

大規模な系の場合、分散できないメモリ量が増大するため、hybrid 並列を用いて 1 MPI プロセスあたりのメモリ量を増やすことが必須になる。

また計算機システムによっては hybrid 並列の方が通信などが高性能になるものもある。また、バンド \times バンド程度の大きさの密行列の固有値問題を解く時、この規模の問題が shared memory 並列では性能が出るが、MPI 並列では性能がでないこともある。ここ部分は高並列では主要な律速の一つなので、高速化が重要である。

hybrid 実行する場合、1 MPI process が 1 つの uniform memory access の単位、通常は 1 chip、を構成するのが通常良い。ただし、非常に高並列になる場合にはもっと大きい単位をとると効果がある場合がある。またこの場合には、FFT ルーチン、LAPACK、密行列対角化ルーチンが shared memory 並列となるようライブラリなどを選択すること。

11.9.1 OpenMP

hybrid 並列を OpenMP によって行っている場合、OpenMP のスレッドに割り当てるスタックの大きさを系の大きさに応じて大きくする必要があることがある。Intel Fortran ではある。この設定は環境変数 OMP_STACKSIZE で行うことができ、OMP_STACKSIZE=256MB と指定すれば 256MB が確保される。なお、各 MPI プロセスに環境変数を渡す方法は MPI の実装によって異なるので注意すること。OpenMPI の場合は mpirun にオプション `-x OMP_STACKSIZE` を渡すことで行える。

11.10 FFTW

FFTW を使っている場合、wisdom はプログラムが実行されるディレクトリにファイル `xtapp_fftw_wisdom.dat` があればそこから読み出される。このファイルはすべての MPI プロセスがアクセスする。またプログラム終了時には wisdom が `xtapp_fftw_wisdom.new.dat` として記録される。

FFTW の初期化ではデフォルトでは詳細にもっとも良い条件を出すように設定してある。そのため第一回目の FFT の初期化には時間がかかることがある。

11.11 処理系による namelist の違い

namelist 中の文字列を引用符でくくらないとならない処理系があるので注意が必要である。gfortran はそのような例である。

11.12 プログラムのテスト例

パッケージ xTAPP-test にはプログラムのテスト例が集められている。この例は同時に実行例としても扱える。Makefile の先頭を調節して make することでテスト例を実行できる。この中にはプログラムの実行スクリプトも含まれており、Fortran の論理機番の割り当ての処理の参考にできる。

11.13 OpenDX による可視化の例

OpenDXによる可視化プログラムの例が xTAPP-test に含まれている。Makefile の TARGET_TOOL で定義されているターゲットを make すると OpenDX 用のデータができるので、これを opendx ディレクトリにある OpenDX のネットワークの例を用いて可視化する。この例の実行には、OpenDX の追加モジュールの CMSP が必要である。これは、Octopus[1] の web ページから入手できる。

[1] <http://www.tddft.org/programs/octopus/>

11.14 プログラムの理解

TAPP-3.0 系列とは元々同じものであったため変数名、サブルーチン名は共通点が多い。そのため TAPP-3.0 系列の文書にある情報は参考になる。

12 動作しないとき

12.1 スタック制限に注意

UNIX 系のシステムでは、標準のスタック制限がきつめにかけてある物がある。このプログラムは OpenMP 並列などのためスタックを多く使うので、この制限を解除しておかないとプログラムが不意に停止することがある。

13 プログラム群が扱う論理機番とファイル

論理機番	内容	標準拡張子
10	入力データ	
34	一番目の擬ポテンシャルデータ。以降 34+ 1 と続く。	
28	一番目の原子電荷データ。以降 28+ 1 と続く。	
18	projected DOS 計算のための動径波動関数データ	pwav
25	電荷密度分布	rho
11	局所ポテンシャル	lpt
95	入力する波動関数	wfn
96	出力する波動関数	wfn
99	構造最適化結果のサマリ	str
50	k 点のトレースを行ったときの固有値データ	band
51	k 点のトレースを行ったときの分布データ	bunpu
57	k 点のトレースを行ったときの軌道電荷密度データ	rok
58	k 点のトレースを行ったときの波動関数データ	wfk
80	STM 像のデータ	stm
55	projected DOS のデータ	bunpu
16	分子動力学計算の軌跡の記録データ	mdlog
51	バンドのコサイン展開係数	pe
73	擬ポテンシャルの平面波展開データ	pots
74	擬ポテンシャルのフィッティングデータ	bpts

論理機番	内容	標準拡張子
75	分子動力学計算の計算継続用の途中状態データ	mdcnt
71	構造最適化の計算継続用の途中状態データ (一次元最適化)	lmd
70	構造最適時の計算継続用の途中状態データ (CG 法)	acgd
72	SCF ループの途中状態データ	scfd
59	四面体法で projected DOS を計算するためのデータ	dosms
113	逆空間表現のワニエ関数データ	rcpw
150	ワニエ関数をブロッホ関数から構成する際の変換行列	amat
122	ワニエ関数によってハミルトニアンを計算したもの	hmatr
116	ワニエ関数の実空間表現のデータ	rwan
134	ワニエ関数の実空間での中心位置	posw

14 ファイルフォーマット

14.1 51 番: k 点のトレースを行ったときの分布データ

以下の Fortran 擬似コードによるアスキーファイルフォーマットが用いられる。

```

if (distrib_mode.eq.'plate') then
  write(6,*) ifbunp=1
else if (distrib_mode.eq.'sphere') then
  write(6,*) ifbunp=2
end if
write(6,*) number_trace_block,number_band_traced,ifbunp,number_spin, &
      number_component
write(6,*) (nkfi(nb),nb=1,number_trace_block+1)
do i=1,number_trace_block+1
  x1=ak(1,i)*bb(1,1)+ak(2,i)*bb(1,2)+ak(3,i)*bb(1,3)
  x2=ak(1,i)*bb(2,1)+ak(2,i)*bb(2,2)+ak(3,i)*bb(2,3)
  x3=ak(1,i)*bb(3,1)+ak(2,i)*bb(3,2)+ak(3,i)*bb(3,3)
  write(6,*) x1,x2,x3
end do
if (distrib_mode.eq.'plate') then
  write(6,*) distrib_number_plate
else if (distrib_mode.eq.'sphere') then
  write(6,*) distrib_number_sphere
end if
icount=0
do nb=1,number_trace_block
  nke=nkfi(nb)
  if(nb.eq.nbk) nke=nke+1
  do nk=1,nke
    sk(1:3)=ak(1:3,nb)+(ak(1:3,nb+1)-ak(1:3,nb))*dble(nk-1)/dble(nkfi(nb))
    icount=icount+1
    write(6,*) icount,sk,distrib_band_lower,distrib_band_upper, &
          number_band_traced
  end do
end do

```



```

do isph=1,nsph
  if (distrib_mode.eq.'plate') then
    write(6,*) iaxis(isph),rmin(isph),rmax(isph),ndiv(isph)
  else if (distrib_mode.eq.'sphere') then
    write(6,*) (rzero(i,isph),i=1,3),rmin(isph),rmax(isph),ndiv(isph)
  end if
  do is=1,number_spin
    do ib=distrib_band_lower,distrib_band_upper
      write(6,*) ib,eigen_energy(ib)
      write(6,*) ((chr(id,idents),id=0,ndiv),idents=1,number_component**2)
    end do
  end do
end do
end do
end do

```

以上の解釈においては inputformat.tex で説明されている入力変数名を参照すること。また

bb(1:3,1:3): 逆格子の基本ベクトル。左添字が x, y, z 。右添字が基本ベクトルの番号。

sk(1:3): 計算している k 点の位置をセル座標で書いたもの

eigen_energy(ib): k 点位置 sk での第 ib バンドの固有値

chr(id,idents): 波動関数分布の値。id は分割インデックス、idents は密度成分の番号。

idents=1 は電荷、idents=2:4 はスピンの Cartesian 座標成分。

内容は distrib_mode によって異なる。

である。